

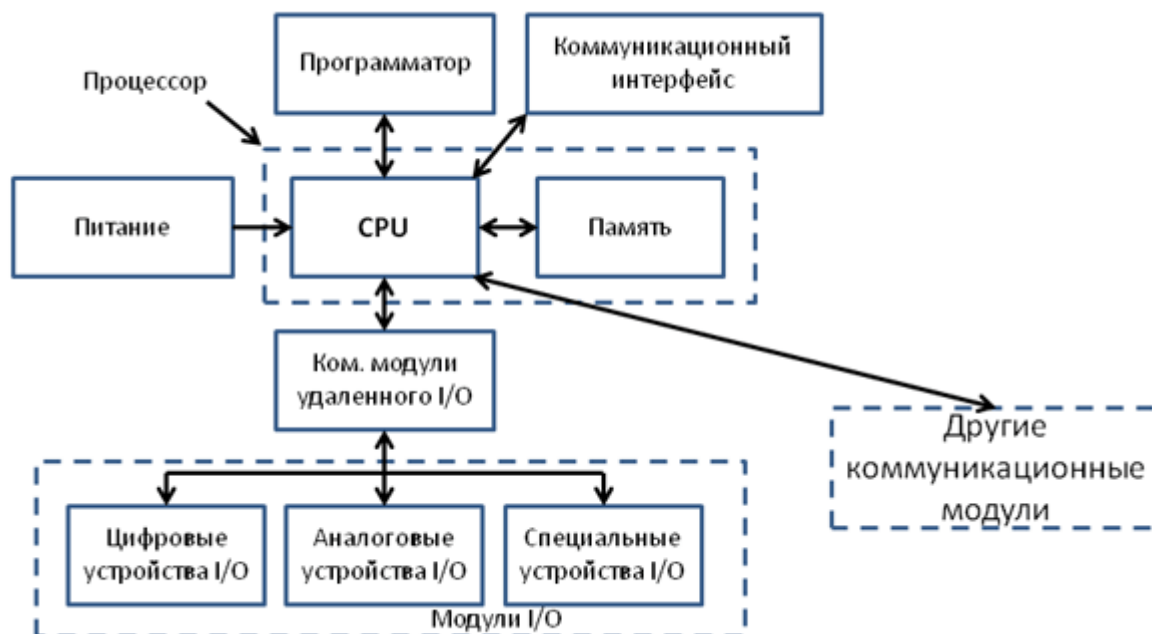
# Архитектура ПЛК. Основы

Февраль 2013

Во многих отношениях, архитектура программируемых логических контроллеров (ПЛК) напоминает архитектуру ПК общего назначения со специализированными модулями ввода/вывода (I/O). Однако некоторые важные характеристики отличают эти устройства. Во-первых, ПЛК намного более надежны и рассчитаны на безотказную работу в течение многих лет – и это их самая важная особенность. Во-вторых, ПЛК могут использоваться в условиях промышленного производства, где им приходится работать в условиях серьезного электромагнитного излучения, вибрации, экстремальных температур и влажности. В-третьих, ПЛК легко обслуживаются техническим персоналом на производстве.

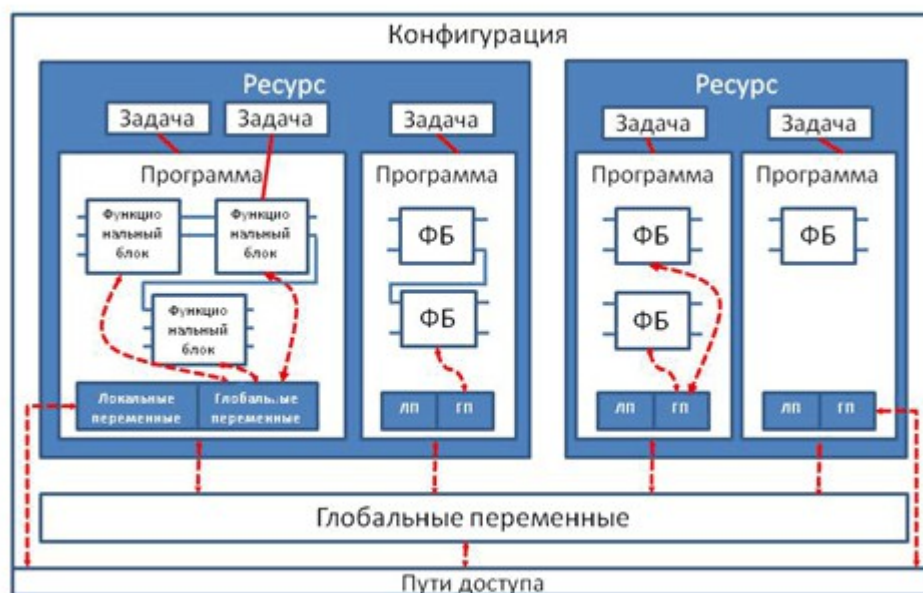
## Аппаратная архитектура ПЛК. Основы

В архитектуре ПЛК мы находим следующие основные компоненты: процессорный модуль, блок питания и модули ввода/вывода. Процессорный модуль состоит из собственно процессора (CPU) и памяти. Кроме того, есть интерфейс к программатору, а также могут быть интерфейсы к удаленным устройствам ввода/вывода и коммуникационным сетям. Питание обычно реализуется с помощью отдельного модуля, как и ввод/вывод. Модули ввода/вывода бывают дискретными (вкл./выкл.), и аналоговыми, а также существуют специализированные модули, например, для управления перемещениями или высокоскоростные счетчики. К модулям ввода/вывода подключаются полевые устройства.



В зависимости от количества каналов ввода/вывода, и конкретного процессора ПЛК, модули ввода/вывода могут быть на том же шасси, что и процессор, и/или на другом или, даже, нескольких шасси. До конца 80-ых, модули ввода/вывода типичных ПЛК размещались на отдельном шасси. В стандартном ПЛК сегодняшнего дня, как правило, модули ввода/вывода расположены на одном шасси с процессором. Некоторые ПЛК позволяют установку больше чем одного процессора на шасси. ПЛК меньшего размера часто монтируются на DIN-рейку. У самых компактных ПЛК, таких как микро-ПЛК или нано-ПЛК, блок питания, процессорный модуль, и весь ввод/вывод идут в одном корпусе. Некоторые микро-ПЛК

оборудуются встроенной панелью операторского интерфейса. Для большинства микро-ПЛК количество каналов ввода/вывода ограничено и не подлежит расширению.



## Программная архитектура (IEC 61131-3)

Стандарт IEC 61131-3 задает программную модель, соответствующую современным принципам разработки программных продуктов. Эта модель характеризуется такими свойствами, как разработка «сверху-вниз», структурированное программирование, иерархическая организация, программные интерфейсы и инкапсуляция. К счастью, для того, чтобы эффективно программировать ПЛК, не нужно длительного обучения; хотя, конечно, будучи полностью реализованной, модель IEC 61131-3 весьма сложна. Это основной недостаток модели, особенно по контрасту с простотой первых ПЛК.

Ниже описывается программная модель IEC 61131-3. Рассмотрим основные элементы подробнее.

**Конфигурация** (configuration) – совокупность всего ПО и данных, относящихся к системе ПЛК. В большинстве случаев, конфигурация относится к программам и данным одного ПЛК. В сложных системах, в которых взаимодействуют много различных ПЛК, у каждого есть своя конфигурация. Конфигурация обменивается данными с другими конфигурациями IEC в системе автоматизации через интерфейсы, называемые **пути доступа** (access paths). Использование слова «конфигурация» в этом контексте приводит к терминологическому конфликту, в связи с изначальным использованием этого термина в отрасли автоматизации как относящегося к таким объектам как модель процессора в ПЛК, коммуникационные интерфейсы, удаленный ввод/вывод, распределение памяти и т.д. Поэтому вендоры, производящие IEC-совместимые ПЛК, используют термин «конфигурация» в историческом значении, и описывают совокупность программ и параметров термином **проект** (project).

**Ресурс** (resource) обеспечивает выполнение программ. Один или несколько ресурсов составляют конфигурацию. Обычно ресурс существует в границах ПЛК, но также он может существовать на ПК, допустим, для программного тестирования. Одна из основных функций ресурса – обеспечивать интерфейс между программой и физическим вводом/выводом ПЛК.

**Программа** (program) обычно состоит из взаимосвязанных **функциональных блоков** (function blocks), каждый из которых может быть написан на любом из языков ИЕС. Функциональные блоки или программы также называются **программными модулями** (program organization unit). Помимо функциональных блоков, в программе декларируются физические вводы/выводы и локальные переменные программы. Программа может считывать/вести запись в каналы ввода/вывода, глобальные переменные, и обмениваться данными с другими программами. Пути доступа позволяют конфигурациям обмениваться данными. **Задачи** (tasks) управляют выполнением программ или частей программ.

Задача управляет выполнением одной или нескольких программ и/или функциональных блоков. Выполнение программы подразумевает, что все функциональные блоки программы обрабатываются один раз. Выполнение функционального блока подразумевает, что все его программные элементы выполняются один раз. Для выполнения программы назначаются задачам, с настройкой для непрерывного, периодического выполнения или выполнения по триггеру (событию).

**Переменные** (Variables) задаются в различных программных элементах модели.

**Локальная переменная** (local variable) определяется программным элементом и доступна только ему. Локальные переменные могут быть заданы для функционального блока, программы, ресурса или конфигурации.

**Глобальная переменная** (global variable) заданная для конфигурации, ресурса или программы, доступна всем элементам, которые в них содержатся. Например, глобальная переменная конфигурации доступна всем программным элементам конфигурации. Глобальная переменная программы доступна всем функциональным блокам программы.

Об авторе:

**Доктор Келвин Т. Эриксон (Kelvin T. Erickson)** – профессор области компьютерной техники и электротехники в Missouri University of Science and Technology. Основные сферы интереса: автоматизация производства и АСУ ТП. Является членом ISA и старшим членом IEEE.